|  |
| --- |
| PS2Win |
| Requirements Analysis Process |
| Keep Your Time |

|  |
| --- |
| Carla Machado  09-03-2013 |

Content

[1. Purpose 1](#_Toc351296686)

[2. Inputs and Outputs 1](#_Toc351296687)

[2.1. Inputs 1](#_Toc351296688)

[2.2. Outputs 1](#_Toc351296689)

[3. Activities 1](#_Toc351296690)

[3.1. Elicit Customer needs 1](#_Toc351296691)

[3.2. Define Assumptions and Restrains 2](#_Toc351296692)

[3.3. Definition of use cases or user stories 2](#_Toc351296693)

[3.4. Specification of System Requirements 2](#_Toc351296694)

[3.4.1. Requirements Analysis 3](#_Toc351296695)

[3.4.2. Requirements Specification 3](#_Toc351296696)

[3.4.3. Updating Requirements 4](#_Toc351296697)

[5. Tools 5](#_Toc351296698)

[6. Related Processes 5](#_Toc351296699)

[7. Measures 6](#_Toc351296700)

**Images**

[Figure 1: Process Flowchart 5](#_Toc351217382)

**Tables**

[Table 1: List of Contribuitors ii](#_Toc349382241)

[Table 2: Version history ii](#_Toc349382242)

|  |  |  |  |
| --- | --- | --- | --- |
| **Authors and Contributors** | | | |
| **Date** | **Name** | **Contacts** | **Contribution** |
| 09-03-2013 | Carla Machado | a21170460@alunos.isec.pt | Author |
| 09-03-2013 | João Girão | a21170831@alunos.isec.pt | Author |
| 16-03-2013 | Rui Ganhoto | a21170262@alunos.isec.pt | Contributor |
| 17-03-2013 | Filipe Brandão | a21108276@alunos.isec.pt | Contributor |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table 1: List of Contributors

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Revision History** | | | | | |
| **Date** | **Description** | **Author** | **Version** | **Approvers** | **State** |
| 09-03-2013 | Creation of first draft | Carla Machado &  João Girão | 0.1 |  | Draft |
| 16-03-2013 | Finishing first draft | Carla Machado &  João Girão | 0.2 |  | Ready for revision |
| 16-03-2013 | Process Review | Rui Ganhoto | 0.2 |  | Ready for Revision |
| 16-03-2013 | Small changes as a result of the review. | Carla Machado | 0.3 |  | Ready for Approval |
| 17-03-2013 | Approved |  | 0.3 | Filipe Brandão |  |
| 20-03-2013 | Changes as a result of the weekly meeting | Carla Machado | 0.4 |  | Draft |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Table 2: Version history

# Purpose

The purpose of the Requirements Analysis Process is the analysis and the detailing of client needs and requirements in order to be able to provide a system requirement specification.

The process also has the purpose of assuring that the customer needs and expectations are understood and the right solution is developed.

# Inputs and Outputs

In this chapter the inputs and outputs of the Requirements Analyzes Process will be described.

# Inputs

This process will be initiated by the opportunity of developing a software project.

The main inputs of the process will be the customer needs including any documentation and other forms of information provided by the customer and if possible user needs. One of the documents that should serve as an input of the process is the Vision & Scope.

Furthermore the input of the project team and any additional stakeholders should be taken into account.

# Outputs

The output of the process will be a word document specifying the system requirements, the System Requirements Specification, and also a number of complementary files such as the project files of the requirements management tool or use case tool.

# Activities

# Elicit Customer needs

The purpose of this activity will be the clarification of the customer needs as well as the gathering of information about the solution to develop.

In this stage there are a number of techniques that can be used. The choice of the ones that should be used must be made according to the specific needs of the project.

Some of the techniques that can be used are:

* Brainstorming;
* Questionnaires, interviews and scenarios;
* Prototypes and models;
* Use cases drafts;
* Workshops with the stakeholders;

The outputs of these activities will be used to the uses case definition and requirements specification.

One of the mandatory outputs of this activity is the Business Rules that should be registered in the project file of Enterprise Architect (EA) and follow the naming convention BR-Number.

# Define Assumptions and Restrains

The purpose of this activity is the definition of the project assumptions and restrains. These assumptions and restrains can be of a technical order or related to the business rules. The result of this activity will be registered in the project file of EA. Each assumption and restrain will be defined by a given name and description. In cases where the assumption or retrains has its origin in the business rules a connection must be made in the EA file to assure traceability.

The naming of the items should follow a convention accorded by the team. A suggestion for the convention to use is

* AS-Number for the assumptions;
* RST-Number for the restrains.

# Definition of use cases or user stories

The purpose of this activity is the definition of the intervenient players, uses cases and the possible scenarios. The use cases will provides a set of scenarios that convey how the system should interact with a human user or another system.

The use cases will be registered in the project file of EA. For each use case will be defined at least a name and a description. Other information that can be defined are the possible paths and the preconditions. Each use cases should be classified according to priority considering which ones the client considers more relevant.

Each use case must be related to the Business Rules or element that originated the use case. Furthermore it must identify the creator of the requirements along with the date of the last update.

# Specification of System Requirements

The purpose of this activity is the specification of the requirements. At this stage and as long as the requirements and System Requirements specifications aren’t baselined the requirements can be updated or deprecated without recourse to a formal process.

For the update of the version requirements only major versions are considered. Requirements version can only be updated on change state for baselined.

## Requirements Analysis

The requirements must be classified by type such as functional or performance or others and also given a level of priority and complexity. The scale to be used is:

* High
* Medium
* Low

The requirements priority attribution must be a result of the client’s concept of which functionalities are more important to the application. As for the complexity it must take into account the input of the development team in order to obtain an understanding of the level of difficulty of implementation.

Furthermore each requirement must be SMART:

* Specific
* Measurable
* Attainable
* Realizable
* Traceable

When conflicting requirements are detected the team alongside with the client must analyze the requirements and implications of the situation in other requirements or elements of the project in order to make a clarification or decision on what requirement should prevail.

In situations in which a consensus couldn’t be reached the client’s point of view should take precedence.

## Requirements Specification

The requirements should be properly identified and categorized by type:

* functional requirements,
* non-functional requirements
* user requirements
* Other types.

The requirements will be registered in the project file of EA and related to the elements that originated the requirement.

The nomination of each requirement must be consistent through them all and must be accorded project by project.

The output of this activity is the exportation of the project file in EA to a word document the Software Requirements Specification (SRS).

## Updating Requirements

Once the requirements and the SRS are baselined any change to them should follow a formal procedure consisting of the following steps:

When a necessity to change the requirements is detected either by the existence of conflicting requirements or changes in the client’s needs or other reasons a requirements change should be purposed.

* The person purposing a requirements change should (using Change Requirement Template):
  + Cleary Identify the motives behind the change
  + Analyze the impact of the change
    - Other requirements or elements that could be affected
    - Effort need for implementing the change and the tasks involved.
  + Present the request change to three people:
    - Project Manager
    - Technical Manager
    - Quality Manager
* The approvers must analyze the information provided and analyze the impact to the project. Decide if the change should proceed or not
* If the change request is approved
  + Requirements change to state “Draft”
  + A person is elected to perform the changes to the requirements file in EA
  + The changes are performed
  + The new requirements are reviewed by a third person
    - Should verify possible conflicts created by the changes
    - Could result in new changes
  + The new requirements are presented for approval
    - The approvers are the same roles that approved the change
* If the changes are approved
  + The requirements in EA are baselined
  + The Document Management should be initiated in order to update the SRS
* If the changes aren’t approved
  + The changes can be rejected and all changes are reversed
    - Requirements revert to last baselined state
  + A new phase of editing the requirements takes place
    1. **Deprecating Requirements**

The activity of deprecating a requirement is similar to the update activity with the distinction that at least some of the requirements involved in the activity will not be baselined but deprecated.

1. **Flow Chart**

In Figure 1 the process flowchart with the various decisions and actions is presented.



Figure 1: Process Flowchart

# Tools

The tools to be used in the activities of this process are a use case and a requirements management tool the Enterprise Architect. For documentation purposes Microsoft Office tools will be used.

# Related Processes

This process is related to the Document Management Process that should be followed when creating and updating the Software Requirements Specification.

# Measures

The measures to consider as result of this process are:

* Level of connectivity of the requirements
  + Percentage of requirements that possess a certain number of connections
* Level of change of the requirements
  + Number of requirements by major version number
  + Percentage of change – Number of alterations/total requirements
* Number of requirements by Functional or Non Functional type